UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/772,533	02/05/2004	Dan Teodosiu	13768.783.7	5407

47973 7590 04/23/2008
WORKMAN NYDEGGER/MICROSOFT
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UT 84111

EXAMINER	
SYED, FARHAN M	

ART UNIT	PAPER NUMBER
2165	

MAIL DATE	DELIVERY MODE
04/23/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	**Application No.** 10/772,533	**Applicant(s)** TEODOSIU ET AL.	
	Examiner FARHAN M. SYED	**Art Unit** 2165	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)□ Responsive to communication(s) filed on *07 May 2007*.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-9 and 23-40* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-9 and 23-40* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application

6) □ Other: _____.

DETAILED ACTION

1. Claims 1-9 and 23-40 are pending.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 1-9 and 23-40 are rejected under 35 U.S.C. 103(a) as being unpatentable over Krein et al (U.S. Patent 6,385,701) in view of a non-patent literature titled "A Scalable Content Distribution Service for Dynamic Web Content" by Seejo Sebastine, University of Virginia, 15 June 2001 (and known hereinafter as Sebastine) and in further view of a non-patent literature titled "Extensible Security Architectures for Java" by Wallach, Dan S., et al, ACM SIGOPS Operating Systems Review, vol. 31, issue 5, December 1997, pages 116-128 (and known hereinafter as Wallach).

As per claims 1, 7, 23, and 37, Krein teaches a computer system for reorganizing storage, comprising (i.e. "A computing environment 100 includes, for instance, at least one computing unit 102 coupled to one or more other computing units 104.")(Column 3, lines 33-35): a file system for receiving a file **access** request (i.e. "File specific server layer 208 is the layer that transforms SMBparser requests into cache and physical file system requests and also interfaces with the DFS token management service to manage the sharing of files between clients.")(Column 4, lines 44-48).

Krien does not explicitly teach a computer system comprising a path rewriter for **prepending another server name to a legacy server name of** a path of a file **access** request received by the file system; and a path redirector for **resolving links to new storage locations which are encountered while traversing the path of file access request**.

Sebstine teaches a computer system comprising a path rewriter for **prepending another server name to a legacy server name of** a path of a file **access** request received by the file system (i.e. "By default, Squid rewrites any HTTP "Host:" headers in redirected requests to point to the new host to which the URL has been redirected,")(Section 4.1.1; Appendex A); and a path redirector for **resolving links to new storage locations which are encountered while traversing the path of file access request** (i.e. "The URL rewriter module handles the job of rewriting the (redirected) URLs that the *active server* receives to fetch the script from the original content server. The address of the content server is specified in the HTTP "Host:" header as elucidated in section 4.1.1. We have used the functionality provided by Apaches *"mod rewrite"* module to perform the URL rewriting. A sample rewrite-rule that can be used is given below: RewriteRule ^/cgi-bin/(.*$) http://%{HTTP_HOST}/cgi-bin/$1 [P]. This rule states that all URLs which are meant to be served from the *cgi-bin* directory are to be rewritten to be served from the same directory, but on the host specified by the HTTP HOST environment variable. Since we wish to cache the script that is returned, we force this request to be a proxy request (indicated by [P]).")(Section 4.2.1).

It would have been obvious to a person of ordinary skill in the art at the time of Applicant's invention to modify the teachings of Krein with the teachings of Sebastine to include a computer system comprising a path rewriter operably coupled to the file system for rewriting a path of a file share request received by the file system; and a path redirector operably coupled to the file system for redirecting the rewritten path of the file share request to another storage location with the motivation to enable clients having different protocols to share the same files (Krein, column 1, lines 39-40).

The combination of Krein and Sebastine do not explicitly teach prepending another server name to a legacy server name.

Wallach teaches prepending another server name to a legacy server name (i.e. "One possible implementation is presented in figure 2. A SubFS represents a capability to access a subtree of the file system. Hidden inside each SubFS is a FileSystem capability; the SubFS prepends a fixed string to all pathnames before accessing the hidden FileSys tern Any code which already possesses a handle to a FileSys tern can create a SubFS, which can then be passed to au untrusted subsystem. Note that a SubFS can also wrap another SubFS, since SubFS implements the same interface as FS.")(page 4).

It would have been obvious to a person of ordinary skill in the art at the time of Applicant's invention to modify the teachings of Krein with the teachings of Sebastine and with the further teachings of Wallach to include prepending another server name to a legacy server name with the motivation to enable clients having different protocols to share the same files (Krein, column 1, lines 39-40).

As per claims 2 and 8, Klien does not explicitly teach a system further comprising a database for storing information about file share requests received by the file system.

Sebastine teaches a system further comprising a database for storing information about file share requests received by the file system (i.e. "The URL manipulations can depend on various tests, for instance server variables, environment variables, HTTP headers, time stamps and even external database lookups in various formats can be used to achieve a really granular URL catching.")(Apendex B).

It would have been obvious to a person of ordinary skill in the art at the time of Applicant's invention to modify the teachings of Krein with the teachings of Sebastine to include a system further comprising a database for storing information about file share requests received by the file system with the motivation to enable clients having different protocols to share the same files (Krein, column 1, lines 39-40).

As per claim 3, Klien does not explicitly teach a system wherein the database comprises a log file

Sebastine teaches a system wherein the database comprises a log file (i.e. "It supports optional logging of URL rewrites to a separate log file, including the number of the rule, which has been used to rewrite the URL.")(Apendex A).

It would have been obvious to a person of ordinary skill in the art at the time of Applicant's invention to modify the teachings of Krein with the teachings of Sebastine to include a system wherein the database comprises a log file with the motivation to enable clients having different protocols to share the same files (Krein, column 1, lines 39-40).

As per claim 4, Klien does not explicitly teach a system wherein the information stored comprises usage information about the file share requests.

Sebastine teaches a system wherein the information stored comprises usage information about the file share requests (i.e. "When an *active server* receives a redirected request, it first checks to see if the script that has been requested is already cached locally in its *script cache*. If so, it executes the script with the current inputs and returns the results to the Squid *proxy cache*.

In case, a cached version of the script is not available locally, the *active server* initiates a URL rewrite operation which is handled by the *URL rewriter* module.")(Section 4.2).

It would have been obvious to a person of ordinary skill in the art at the time of Applicant's invention to modify the teachings of Krein with the teachings of Sebastine to include a system wherein the information stored comprises usage information about the file share requests with the motivation to enable clients having different protocols to share the same files (Krein, column 1, lines 39-40).

As per claim 5, Krein teaches a system wherein the file system comprises a distributed file system (i.e. "For example, data is shared amongst Server Message Block (SMB) clients, (e.g., Windows, OS/2 clients), local users and/or remote Unix clients, such as Distributed File Services for the Distributed Computing Environment (DFS/DCE) clients.")(Column 3, lines 25-30).

As per claim 6, Klien does not explicitly teach a system wherein redirecting the rewritten path of the file share request to another storage location comprises redirecting the rewritten path of the file share request to a storage location on the computer system.

Sebastine teaches a system wherein redirecting the rewritten path of the file share request to another storage location comprises redirecting the rewritten path of the file share request to a storage location on the computer system (i.e. "This rule states that all URLs which are meant to be served from the *cgi-bin* directory are to be rewritten to be served from the same directory, but on the host specified by the HTTP HOST environment variable. Since we wish to cache the script that is returned, we force this request to be a proxy request (indicated by [P]). This proxy request is handled by Apache's *"mod proxy"* module. For example, if a request meant for: http://www.xyz.com/cgi-bin/hello.cgi was redirected to the *active server* at: http://as1.cs.virginia.edu:8080/

then the *active server* on determining that the URL was for a script *hello.cgi* which is not cached locally, rewrites the URL again to: http://www.xyz.com/cgi-bin/hello.cgi and caches the returned results. More extensive details on the syntax can be found in Appendix B.")(Section 4.2.1).

It would have been obvious to a person of ordinary skill in the art at the time of Applicant's invention to modify the teachings of Krein with the teachings of Sebastine to include a system wherein redirecting the rewritten path of the file share request to another storage location comprises redirecting the rewritten path of the file share request to a storage location on the computer system with the motivation to enable clients having different protocols to share the same files (Krein, column 1, lines 39-40).

As per claim 9, Klien teaches a computer readable medium having computer-executable components comprising the system of claim 1 (i.e. "The media has embodied therein, for instance, computer readable program code means for providing and facilitating the capabilities of the present invention.")(Column 19, lines 35-37).

As per claim 24, Krein teaches a method further comprising resolving an aliased legacy server name to establish a connection to the network address of a server (i.e. "Input to the routine is a file id, the address of the specific host (hs_host) of the desired file and the rights desired for the file.")(Column 14, lines 36-38).

As per claim 25, Krein teaches a method further comprising sending an access request to a server for a relocated legacy share path name (i.e. "File specific server layer 208 is the layer that transforms SMBparser requests into cache and physical file system requests and also interfaces with the DFS token management service to manage the sharing of files between clients.")(Column 4, lines 44-48).

As per claim 26, Krein teaches a method wherein sending an access request to a server comprises sending a Dfs create request (i.e. "DFS/DCE client software provides caches of files and directories to avoid network traffic. Additionally, they cache byte range locks, file opens and closes. To facilitate this, a token management scheme is used. If a client has a token with the needed rights for a file or directory, it can cache the data for that file or directory. It obtains these tokens from the central token manager (e.g., DFS token manager 214) residing at the file server.")(Column 5, lines 6-13).

As per claim 27, Krein does not explicitly teach a method wherein rewriting the legacy share path comprises invoking a path rewriter to rewrite the legacy share path.

Sebastine teaches a method wherein rewriting the legacy share path comprises invoking a path rewriter to rewrite the legacy share path (i.e. "By default, Squid rewrites any HTTP "Host:" headers in redirected requests to point to the new host to which the URL has been redirected,")(Section 4.1.1; Appendex A).

It would have been obvious to a person of ordinary skill in the art at the time of Applicant's invention to modify the teachings of Krein with the teachings of Sebastine to include a method wherein rewriting the legacy share path comprises invoking a path rewriter to rewrite the legacy share path with the motivation to enable clients having different protocols to share the same files (Krein, column 1, lines 39-40).

As per claim 28, Krein does not explicitly teach a method further comprising encountering a link while traversing the rewritten legacy share path.

Sebastine teaches a method further comprising encountering a link while traversing the rewritten legacy share path (i.e. “Each script must be linked with this library in order to function correctly.”” The Rewrite Rules Configuration File specifies a regular expression based pattern, which if matched results in the rest of the rule being executed. Just as in the Access Control List, the rules are matched in a linear order and hence the order of the rules is important.”)(Section 4.2.4; Appendex A.4).

It would have been obvious to a person of ordinary skill in the art at the time of Applicant’s invention to modify the teachings of Krein with the teachings of Sebastine to include a method further comprising encountering a link while traversing the rewritten legacy share path with the motivation to enable clients having different protocols to share the same files (Krein, column 1, lines 39-40).

As per claims 29 and 38, Krein does not explicitly teach a method wherein resolving any links in the rewritten legacy share path comprises invoking a path redirector to resolve any links in the rewritten legacy share path.

Sebastine teaches a method wherein resolving any links in the rewritten legacy share path comprises invoking a path redirector to resolve any links in the rewritten legacy share path (i.e. “Each script must be linked with this library in order to function correctly.” “The Rewrite Rules Configuration File specifies a regular expression based pattern, which if matched results in the rest of the rule being executed. Just as in the Access Control List, the rules are matched in a linear order and hence the order of the rules is important.”)(Section 4.2.4; Appendex A.4).

It would have been obvious to a person of ordinary skill in the art at the time of Applicant’s invention to modify the teachings of Krein with the teachings of Sebastine to include a method wherein resolving any links in the rewritten legacy share path comprises invoking a path redirector to resolve any links in the rewritten legacy share path with the motivation to enable clients having different protocols to share the same files (Krein, column 1, lines 39-40).

As per claims 30 and 40, Krein teaches a method further comprising accessing the share path of the storage location of the relocated legacy share (i.e. "The method includes, for instance, accessing a file by a first client of the computing environment, the first client having a first protocol; and accessing the file by a second client of the computing environment, the second client having a second protocol. The second protocol is different from the first protocol, and wherein a change to the file by one of the first client and the second client is reflected to the other of the first client and the second client, thereby providing cache consistency.")(Column 2, lines 1-11).

As per claim 31, Krein teaches a method wherein accessing the share path of the storage location of the relocated legacy share comprises sending a Dfs create request to the network address of the storage location of the relocated legacy share (i.e. "DFS token manager 214 is the layer of the file server used to manage the tokens needed by the clients to access files. For example, DFS/DCE clients obtain tokens before performing an operation locally. That is, they use tokens to allow them to cache data, status and byte range locks without requiring a communication (e.g., a remote procedure call (RPC)) with the server; thus, reducing network traffic and increasing the access speed to remote files. A token represents a client's right to cache the data and it may represent its right to perform an operation.") (Column 4, lines 62-67).

As per claims 32 and 39, Krein teaches a method of claim 30 wherein accessing the share path of the storage location of the relocated legacy share comprises accessing a path of a separate Dfs namespace (i.e. "DFS token manager 214 is the layer of the file server used to manage the tokens needed by the clients to access files. For example, DFS/DCE clients obtain tokens before performing an operation locally. That is, they use tokens to allow them to cache data, status and byte range locks without requiring a communication (e.g., a remote procedure call (RPC)) with the server; thus, reducing network traffic and increasing the access speed to remote files. A token represents a client's right to cache the data and it may represent its right to perform an operation.") (Column 4, lines 62-67).

As per claim 33, Krein teaches a method further comprising encountering a Dfs reparse point while traversing the rewritten legacy share path (i.e. "DFS token manager 214 is the layer of the file server used to manage the tokens needed by the clients to access files. For example, DFS/DCE clients obtain tokens before performing an operation locally. That is, they use tokens to allow them to cache data, status and byte range locks without requiring a communication (e.g., a remote procedure call (RPC)) with the server; thus, reducing network traffic and increasing the access speed to remote files. A token represents a client's right to cache the data and it may represent its right to perform an operation.") (Column 4, lines 62-67).

As per claim 34, Krein teaches a method further comprising returning a message to the client indicating the path contains a link (i.e. "DFS token manager 214 is the layer of the file server used to manage the tokens needed by the clients to access files. For example, DFS/DCE clients obtain tokens before performing an operation locally. That is, they use tokens to allow them to cache data, status and byte range locks without requiring a communication (e.g., a remote procedure call (RPC)) with the server; thus, reducing network traffic and increasing the access speed to remote files. A token represents a client's right to cache the data and it may represent its right to perform an operation.") (Column 4, lines 62-67).

As per claim 35, Krein teaches a method further comprising receiving a referral request message from the client for the referral path (i.e. "DFS token manager 214 is the layer of the file server used to manage the tokens needed by the clients to access files. For example, DFS/DCE clients obtain tokens before performing an operation locally. That is, they use tokens to allow them to cache data, status and byte range locks without requiring a communication (e.g., a remote procedure call (RPC)) with the server; thus, reducing network traffic and increasing the access speed to remote files. A token represents a client's right to cache the data and it may represent its right to perform an operation.") (Column 4, lines 62-67).

As per claim 36, Krein teaches a computer readable medium having computer-executable instructions for performing the method of claim 23 (i.e. "The media has embodied therein, for instance, computer readable program code means for providing and facilitating the capabilities of the present invention.")(Column 19, lines 35-37).

Response to Remarks/Argument

4. Applicant's arguments, see page 8, filed 07 May 2007, with respect to Figure 1, item 192 and 193 have been fully considered and are persuasive. The objection of non-final office action, mailed 30 October 2006, has been withdrawn.

5. Applicant's arguments, see page 8, filed 07 Mail 2007, with respect to claims 1 and 7 have been fully considered and are persuasive. The 35 U.S.C. 112, second paragraph, rejection of the non-final office action has been withdrawn.

6. Applicant's arguments, see 8, filed 07 May 2007, with respect to claims 9 and 36 have been fully considered and are persuasive. The 35 U.S.C. 101 rejection of the non-final office action has been withdrawn.

7. Applicant's arguments with respect to claims 1-9 and 23-40 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

8. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Contact Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Farhan M. Syed whose telephone number is 571-272-7191. The examiner can normally be reached on 8:30AM-5:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Christian Chace can be reached on 571-272-4190. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/F. M. S./
Examiner, Art Unit 2165

/Christian P. Chace/

Supervisory Patent Examiner, Art Unit 2165